Java Programming

What is Java Programming?

- Java is an Object-Oriented, Class-based, Concurrent, Secured and General-Purpose computer Programming Language.
- Java is a programming language and a platform.
- Developed by Sun Microsystems in the year 1995.

Java Applications

- Mobile Applications
- Desktop Applications
- Web Applications
- Web Servers and Application Servers
- Games
- Database Connection
- Embedded System
- Smart Card
- Robotics

Why use Java?

- It works on different platforms.
- It is one of the most popular programming languages in the world.
- It is easy to learn and simple to use.
- It is open source and free.
- It is secure, fast and powerful.
- It has huge community support.
- It is Object Oriented Language.

Java Install

To check if you have Java installed on a Windows PC, search in the start bar for Java or type the following in Command Prompt (cmd.exe):

```
C:\Users\Your Name>java -version
```

If Java is installed, you will see something like this (depending on version):

```
java version "11.0.1" 2018-10-16 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.1+13-LTS)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.1+13-LTS, mixed mode)
```

Java Example

Save the code in Notepad as "Main.java". Open Command Prompt (cmd.exe), navigate to the directory where you saved your file, and type "javac Main.java":

```
public class Main {
  public static void main(String[] args) {
    System.out.println("Hello World");
  }
}
```

```
C:\Users\Your Name>javac Main.java
```

Cont...

This will compile your code. If there are no errors in the code, the command prompt will take you to the next line. Now, type "java Main" to run the file:

```
C:\Users\Your Name>java Main
```

The output should read:

```
Hello World
```

Java Platforms / Editions

- Java SE (Java Standard Edition)
- Java EE (Java Enterprise Edition)
- Java ME (Java Micro Edition)
- Java FX

Features of JAVA

- Simple
- Object-Oriented
- Portable
- Platform Independent
- Secured
- Robust
- Architecture neutral
- Interpreted
- High Performance
- Multithreaded
- Distributed
- Dynamic

C++ Vs Java

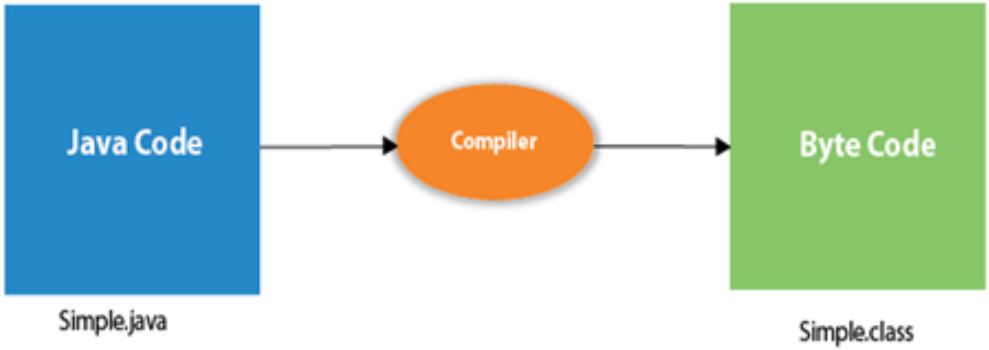
C++	Java
Platform dependent	Platform Independent
Mainly used for system programming	Mainly used for application programming
Supports the goto statement	Doesn't support goto statement
Support Multiple Inheritance	Doesn't support Multiple Inheritance
Supports Operator Overloading	Doesn't support Operator Overloading
Used Compiler only	Used Compiler and Interpreter.

Cont...

C++	Java
Supports both call by value and call by reference,	Supports only call by value.
Supports Structures and Unions	Doesn't supports Structures and Unions.
Doesn't support thread functionality	Support thread functionality
Supports virtual keword	Doesn't support virtual keyword.

Internal Details of Hello Java Program

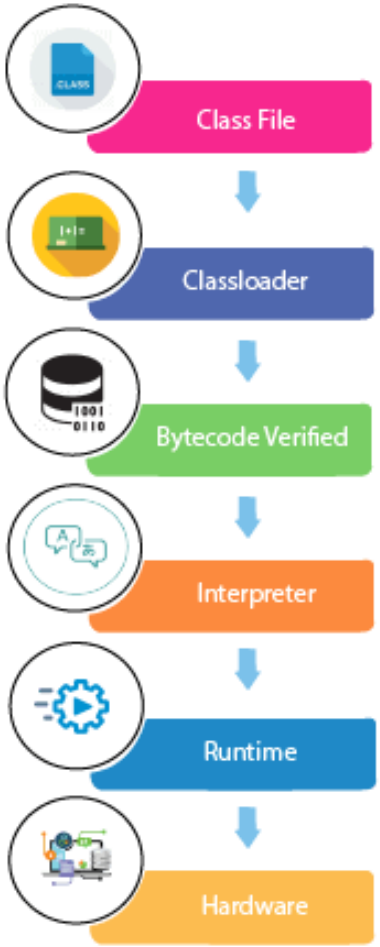
What happens at compile time?



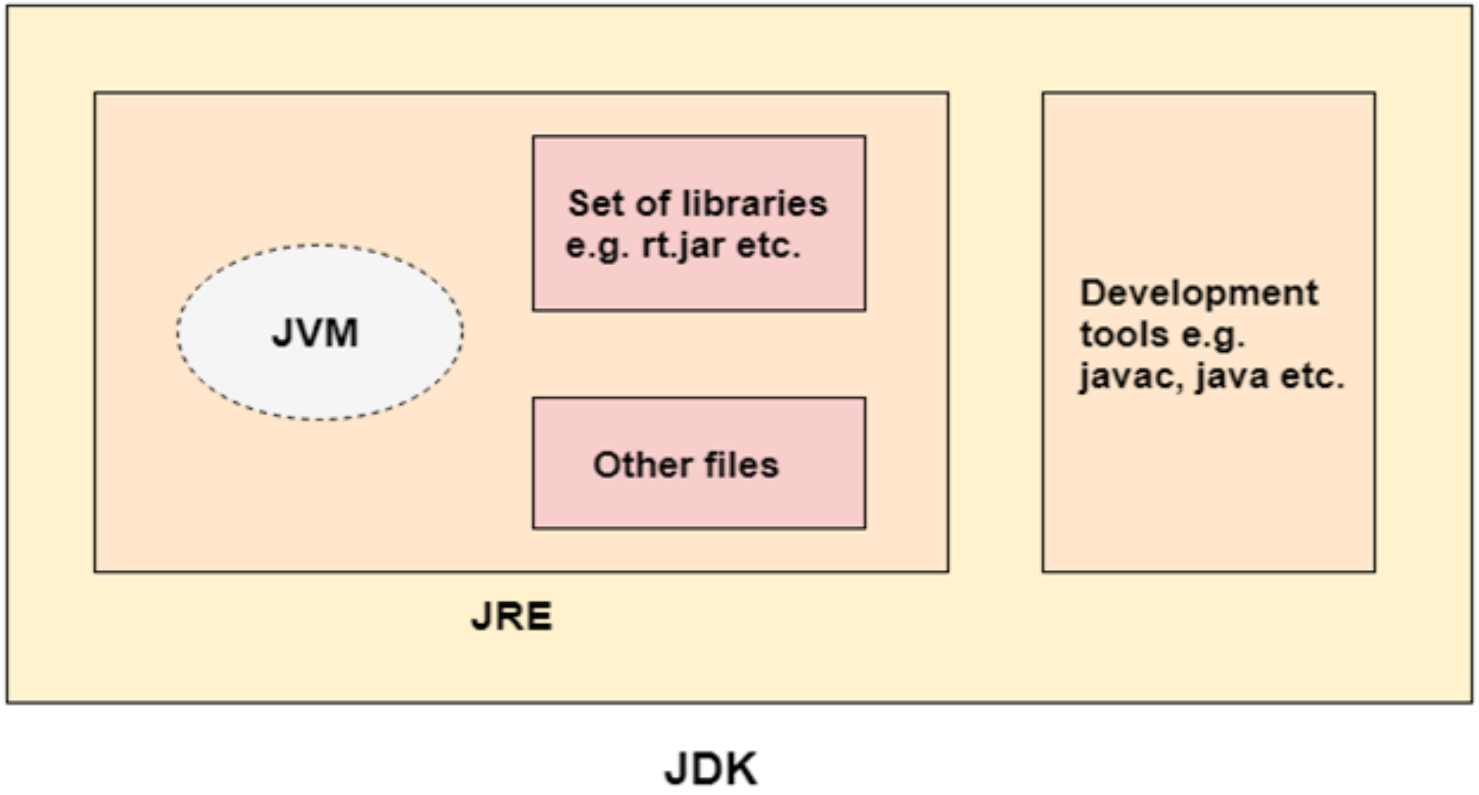
At compile time, java file is compiled by Java Compiler (It does not interact with OS) and converts the java code into bytecode.

Cont...

What happens at runtime?



JVM, JRE, JDK

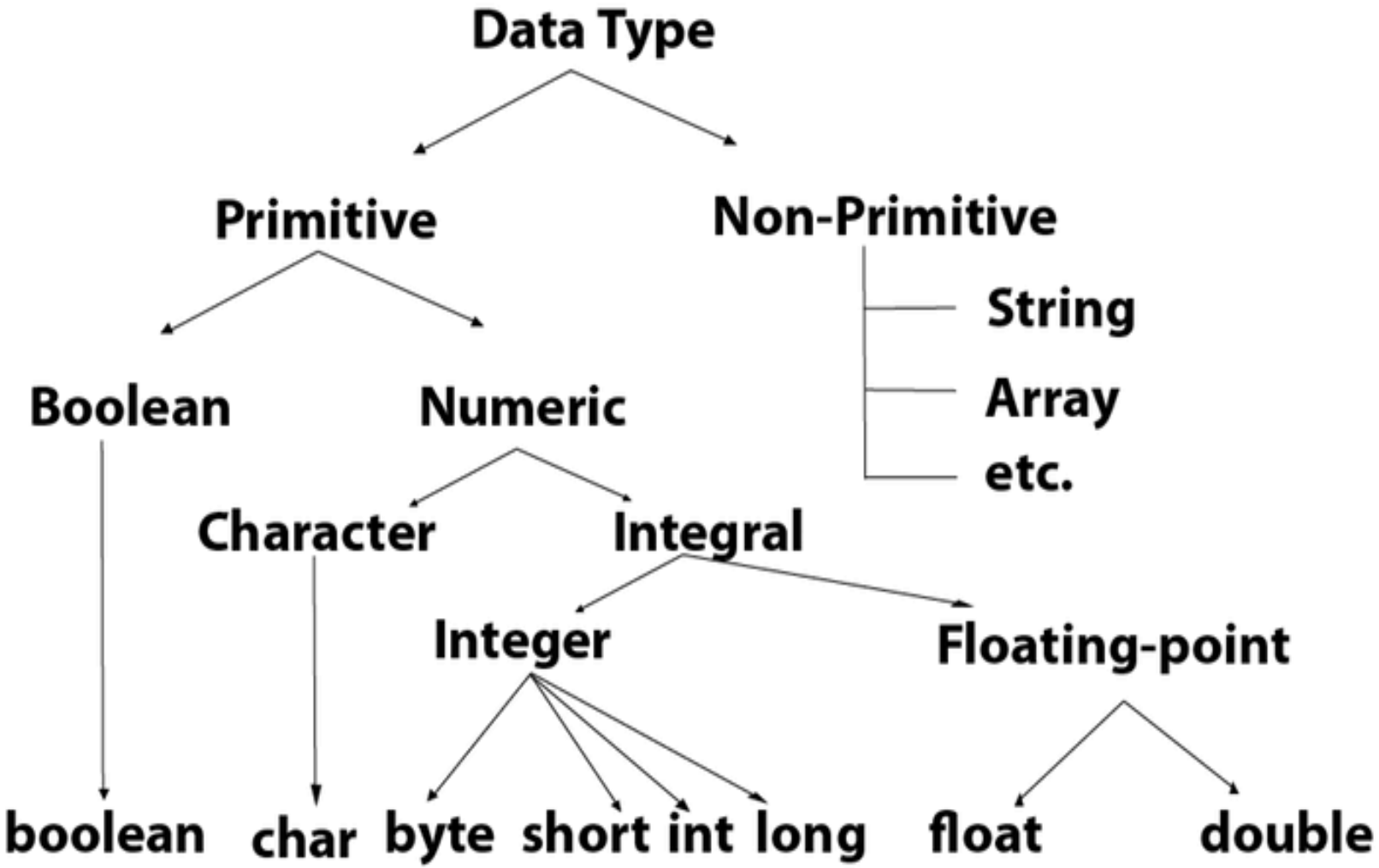

Set of libraries
e.g. rt.jar etc.
JVM
Other files
Development
tools e.g.
javac, java etc.
JRE
JDK

Java Variables

Variables are containers for storing data values.

In Java, there are different types of variables, for example:

- `String` - stores text, such as "Hello". String values are surrounded by double quotes
- `int` - stores integers (whole numbers), without decimals, such as 123 or -123
- `float` - stores floating point numbers, with decimals, such as 19.99 or -19.99
- `char` - stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes
- `boolean` - stores values with two states: true or false


Data Type
Primitive
Non-Primitive
String
Array
etc.
Boolean
Numeric
Character
Integral
Integer
Floating-point
boolean
char
byte
short
int
long
float
double

Java Operators

- Arithmetic Operators
- Assignment Operators
- Relational Operators
- Logical Operators
- Unary Operators
- Bitwise Operators